DOCUMENT RESUME

ED 116 660 IR 002 903

TITLE Soloworks Newsletters Number 32 and 33.
INSTITUTION Pittsburgh Univ., Pa. Project Solo.
SPONS AGENCY National Science Foundation, Washington, D.C.
PUB DATE Dec 75
NOTE 12p.

EDRS PRICE MF-$0.76 HC-$1.58 Plus Postage
DESCRIPTORS *Computer Based Laboratories; Computer Graphics; Mathematics Curriculum; *Mathematics Instruction; Newsletters; *Secondary School Mathematics
IDENTIFIERS Project Solo; *Soloworks; Student Controlled Computing

ABSTRACT

Two issues of the Soloworks newsletter contain information about the Soloworks project, computer equipment, and the educational philosophy that underlies the student-controlled computer based approach to secondary school mathematics instruction. (JY)

SOLOWORKS NEWSLETTER No. 32

PROJECT SOLO, UNIVERSITY OF PITTSBURGH, PITTSBURGH, PA 15260 (412) 624-6461
(The following article is reprinted from the AEDS Journal, Spring, 1975.)

Computers and the Curriculum Question - Project Solo

THOMAS A. DWYER
University of Pittsburgh

Introduction

Educators who haven't used computers but who are giving some thought to the possibility sometimes ask: "But aren't these gadgets just one more example of oversold technology? Aren't we headed for the same letdown that followed the promises of teaching machines, films, language labs and educational television?" This is a good question, deserving of more than a "but this time things will be different" kind of response.

There are actually three kinds of issues that need to be addressed. First, we need to ask whether there are compelling theoretical reasons for believing that future evaluation of how well computers promote learning will be favorable. Then we should look for "behavioral" evidence that says this theory is on the right track. Finally, we must ask the tough practical question: after the honeymoon is over, will computers promote substantial improvements in that variably defined, but all powerful force in schools called "curriculum"?

I believe that there are good answers to the first two questions, but that the third issue is wide open. In this paper I'll first summarize those answers to the first two questions which seem to be not only standing the test of time but which are also proving to be "extensible" ideas. I'll then discuss the curriculum question at somewhat greater length. Here the picture will be less clear. However, there do seem to be three options coming into focus. The form of these options will be discussed, with the goal of encouraging local school experimentation.

The Bigger Picture

The framework within which these remarks are made is the following. The word "computing" is used to roughly mean "instructional computing" as opposed to "administrative computing." I realize that there are grey areas, and that the distinction isn't always that simple. We handle this difficulty in our work at Project Solo (1), not by excluding administrative computing, but by emphasizing technology that supports an idealistic rationale for technological experimentation: ". . . to develop powerfully different educational tools, using artifacts and technology from the computer age in a setting defined by the possible more than the habitual."

Real-school users would not be as free to concentrate on the "what is possible" part of this rationale than would a research project, but the same approach to setting priorities can be used: define a rationale, and then judge appropriateness of use against that standard.

Good ideas for using computers to enhance learning are coming out of the schools at a rate (witness the AEDS programming contest) that suggests many

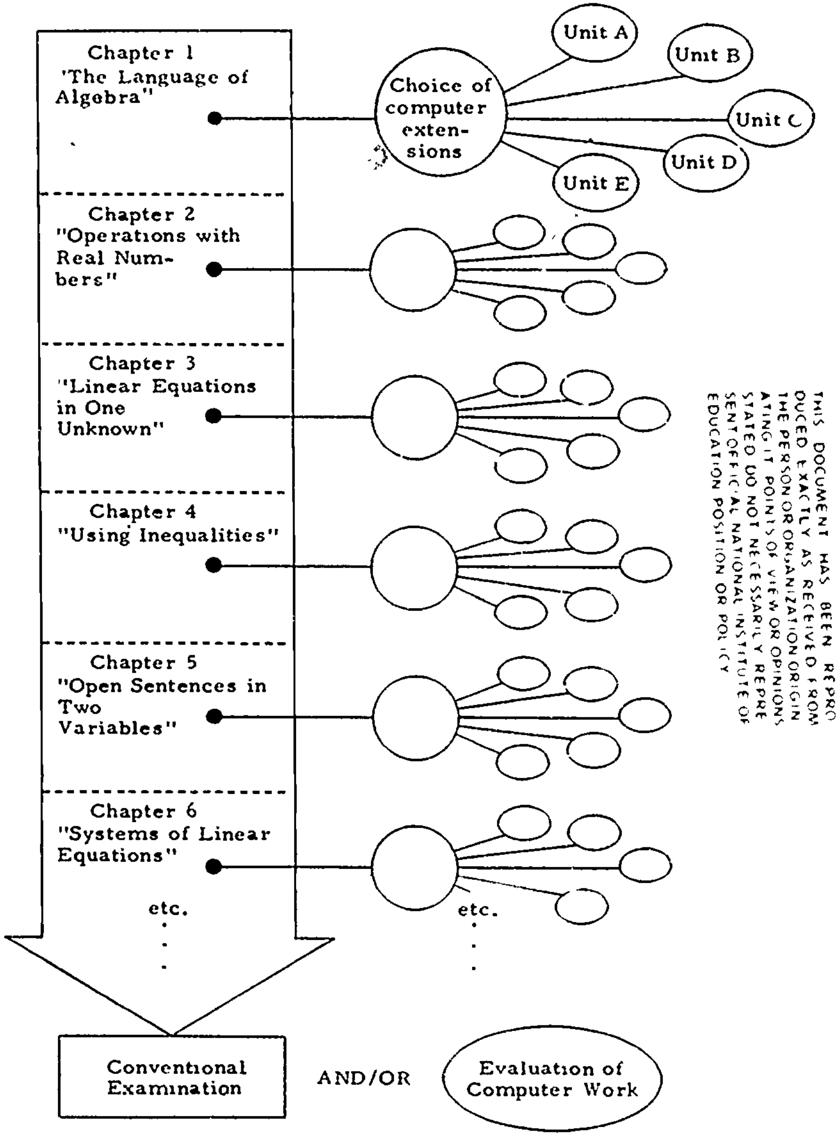

Chapter 1
"The Language of Algebra"
Choice of computer extensions
Unit A
Unit B
Unit C
Unit D
Unit E
Chapter 2
"Operations with Real Numbers"
Chapter 3
"Linear Equations in One Unknown"
Chapter 4
"Using Inequalities"
Chapter 5
"Open Sentences in Two Variables"
Chapter 6
"Systems of Linear Equations"
etc.
etc.
Conventional Examination
AND/OR
Evaluation of Computer Work

Fig. 1. Scheme for attaching computer curriculum units to algebra course

educators have developed such a rationale, and that it includes a strong instructional component. I will proceed on the assumption that similar instructional considerations will be foremost in the planning of new educational computer installations, whether based on large multi-user systems, or the new desk-top microcomputer technology (2). In particular, I will also assume that the instructional use of computers will encompass a sizeable effort to support students in actually programming the computer, for the reasons given by Luehrmann in (3).

Theoretical Promise of Computers for Learning

Even with the preceding assumptions, the potential of computer technology depends entirely on how it is used. Use of the computer to emulate existing educational practices holds only minor promise of making any change. To take an extreme (but actual) example, using a computer to print selected pages from a book misses the point of the machine's potential badly. It doesn't understand the distinction between using technology to *transmit* knowledge (e.g., television), and using technology to create highly interactive environments where new ideas can be *experienced* or even *created*. Computers have unique potential in these last two categories, and that's where their real theoretical promise lies. Until this potential is fully exploited, evaluation is going to be deceptive.

A fuller discussion of a "theory" of computer-enhanced learning based on the distinctions between transmittal, experiential and creative modes of learning is given in (4). An analysis of why the most interesting aspects of CAI are tied to the idea of highly interactive systems is given in Nievergelt (5).

Behaviors Unique to Computer Supported Learning

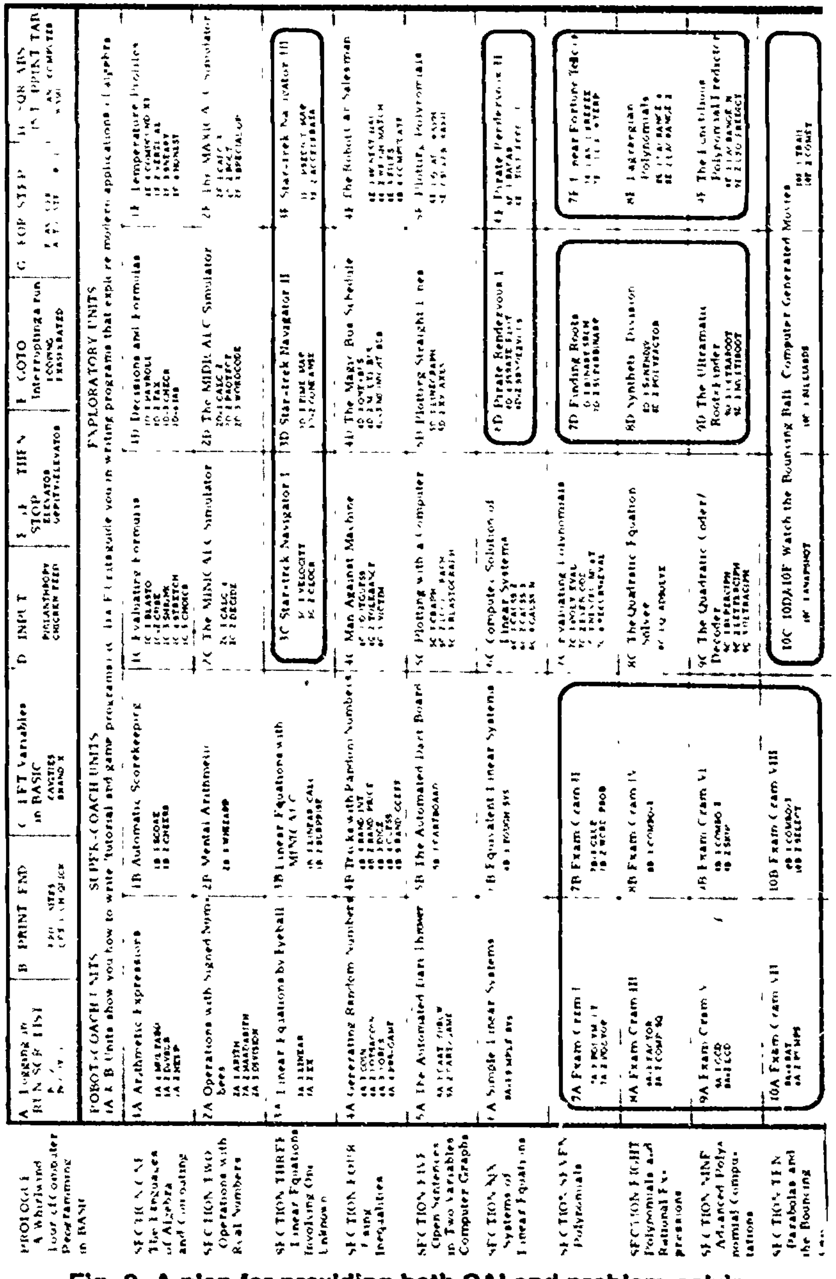
In analyzing the potential of computers to support learning, one is forced to make comparisons. This is done to prove the uniqueness of the computer tool, not to berate the other technology. There have been, and will continue to be, brilliant uses of film, television and other technologically oriented media. But that doesn't change the fact that computing is different, very different.

The truth of this statement is easily demonstrated by considering questions of the following kind: must students be continually "thrown out" at night because they wanted just one more hour on the A-V teaching machine? Do language lab students regularly create new tape programs for their own and other's use? Do the viewers of educational TV swamp the station with new proofs of theorems? The answer to each of these questions when the phrase "interactive computing" is used in appropriate places, and other obvious changes are made, is definitely yes. Computing does motivate students to work endless hours, these students do write new and ingenious programs, and creativity (including new proofs) can be looked for as a regular feature of an interactive computing environment. For more on this subject see Koetke (6), Burleigh (7) and Bell (8).

The Curriculum Question

This is the tough one. It is an especially difficult problem in the area of mathematics education, due in part to the disillusionment of many with the "new math." Critchfield (9) traces the historical and social origins of the prob-

NOTE: A full course of studies based on the units shown in Figure 2 can be found in A Computer Resource Book: Algebra (Houghton-Mifflin Co., One Beacon Street, Boston, Mass. 02107)



PROLOGUE A Whirlwind Tour of Computer Programming in BASIC	A Logging in RUN SCR LIST	B PRINT END	C LET Variables in BASIC CAVITIES BRAND X	D INPUT PHILANTHROPY CHICKEN FEED	E IF THEN STOP ELEVATOR UPPITY-ELEVATOR	F GOTO Interrupting a run LOOPING FRASTRATED	G FOR STEP	H SQR ABS INT PRINT TAB
	ROBOT-COACH UNITS	SUPER-COACH UNITS			EXPLORATORY UNITS			
	(A & B Units show you how to write 'tutorial and game programs'		[illegible]		guide you in writing programs that explore modern applications of algebra			
SECTION ONE The Language of Algebra and Computing	1A Arithmetic Expressions	1B Automatic Scorekeeping		1C Evaluating Formulas		1D Decisions and Formulas		1E Temperature Profiles
SECTION TWO Operations with Real Numbers	2A Operations with Signed Numbers	2B Mental Arithmetic		2C The MINICALC Simulator		2D The MIDICALC Simulator		2E The MAXICALC Simulator
SECTION THREE Linear Equations Involving One Unknown	3A Linear Equations by Eyeball	3B Linear Equations with MINICALC		3C Star-trek Navigator I		3D Star-trek Navigator II		3E Star-trek Navigator III
SECTION FOUR Using Inequalities	4A Generating Random Numbers	4B Tricks with Random Numbers		4C Man Against Machine		4D The Magic Bus Schedule		4E The Robot Car Salesman
SECTION FIVE Open Sentences in Two Variables Computer Graphs	5A The Automated Dart Thrower	5B The Automated Dart Board		5C Plotting with a Computer		5D Plotting Straight Lines		5E Plotting Polynomials
SECTION SIX Systems of Linear Equations	6A Simple Linear Systems	6B Equivalent Linear Systems		6C Computer Solution of Linear Systems		6D Pirate Rendezvous I		6E Pirate Rendezvous II
SECTION SEVEN Polynomials	7A Exam Cram I	7B Exam Cram II		7C Evaluating Polynomials		7D Finding Roots		7E Linear Fortune Teller
SECTION EIGHT Polynomials and Rational Expressions	8A Exam Cram III	8B Exam Cram IV		8C The Quadratic Equation Solver		8D Synthetic Division		8E Lagrangian Polynomials
SECTION NINE Advanced Polynomial Computations	9A Exam Cram V	9B Exam Cram VI		9C The Quadratic Coder/Decoder		9D The Ultramatic Root-Finder		9E The Continuous Polynomial Predictor
SECTION TEN Parabolas and the Bouncing Ball	10A Exam Cram VII	10B Exam Cram VIII		10C 10D&10E Watch the Bouncing Ball: Computer Generated Movies				

Fig. 2. A plan for providing both CAI and problem-solving units within an algebra syllabus

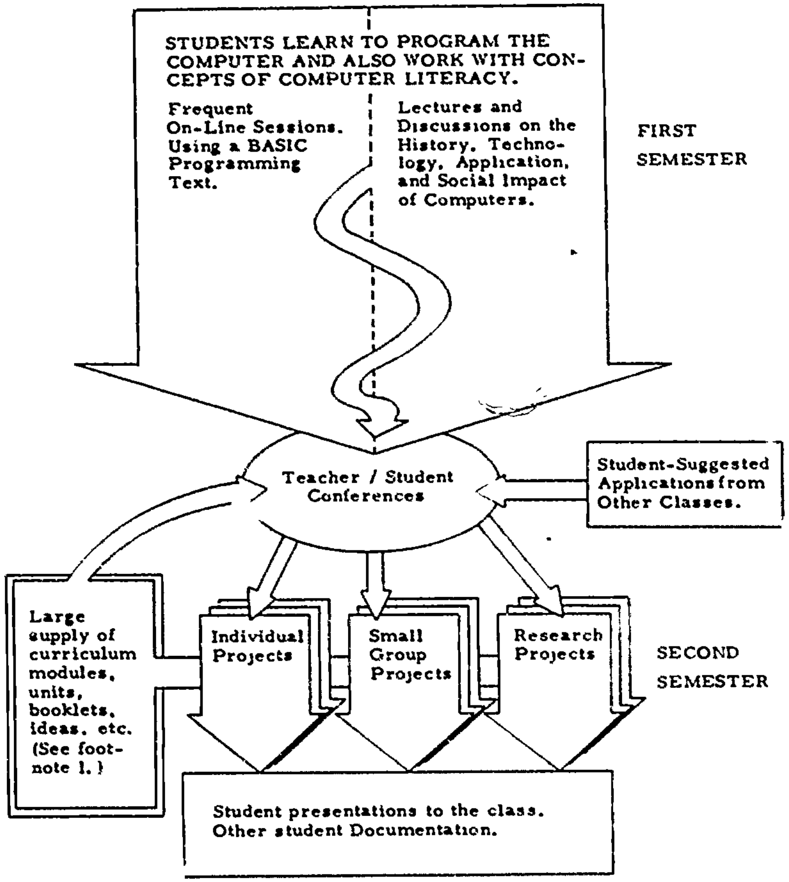

STUDENTS LEARN TO PROGRAM THE COMPUTER AND ALSO WORK WITH CONCEPTS OF COMPUTER LITERACY.
Frequent On-Line Sessions. Using a BASIC Programming Text.
Lectures and Discussions on the History, Technology, Application, and Social Impact of Computers.
FIRST SEMESTER
Teacher / Student Conferences
Student-Suggested Applications from Other Classes.
Large supply of curriculum modules, units, booklets, ideas, etc. (See footnote 1.)
Individual Projects
Small Group Projects
Research Projects
SECOND SEMESTER
Student presentations to the class. Other student Documentation.

Fig. 3. Scheme for a new course in computer science or computer literacy

lem, and notes that educators today are much more likely to pick and choose when it comes to establishing a curriculum.

Much computer activity is tied to mathematics courses; fortunately, the approach to curriculum of the computer-oriented educators is a refreshing break with the formalism of the new math movement. There have been new contributions to both the content (10) and style (11) of school math from computer users and authors. Other traditional academic fields have not been as heavily influenced yet, but there are signs that this situation is changing.

There seem to be three options available to the school that wants to bring computing and computer-enhanced learning into its instructional program: (1) attach use of the computer to a conventional syllabus; (2) create a new course, and (3) "other." I'll discuss each of these in turn.

Using the Computer Within a Conventional Curriculum

This is the most "natural" of the options, but it's the one that needs most care. It is easy to make a flop of this approach by falling into the mistake mentioned earlier, namely, using the computer to emulate old practices. In particular, the computer should not be used to handle the same problems that have traditionally been done by hand. Also, some variety should be allowed; not taking advantage of the computer's flexibility is to miss its potential for handling individual differences. Figure 1 shows a scheme for attaching several different kinds of units to a conventional algebra course.

Diversity can be handled by including CAI drill and practice units, as well as simulation and problem solving units. However, the CAI units can be written in BASIC so that students can modify and/or improve upon this collection. This idea works very well in practice (12). Figure 2 shows a collection of such units keyed to algebra chapter headings. The program "code" names give some suggestion of the content.

Using the Computer in a New Course

This approach has the advantage of not being constrained by previous notions of how a course should be taught, or what material should be covered. The most popular names for such a course are "Computer Science," "Computer Literacy," and "Computer Mathematics." This last title is often used more to satisfy the requirement of giving a math credit than to indicate content.

Since good textbooks for such a course are hard to find, many schools combine a book on programming with various collections of modular materials[1]. When hands-on computing is a part of such a course, it becomes very popular with students. A diagram showing a possible organization for such a two-semester course is shown in Figure 3.

Other Possibilities

This category is meant to leave room for ideas that don't fall into the usual format of scheduled classes. One of the more interesting possibilities is the community learning center. (The People's Computer Co., Box 310, Menlo Park, CA 94025 is one of the best known examples.)

I believe that the advent of more complicated and esoteric hardware (13) makes the idea of a technology oriented learning center which exists outside the normal school an attractive way to go *temporarily*. Many schools could use this center, without commiting themselves to big expenditures, until they had a chance to look the idea over. However, I believe that it would be important to release teachers as well as students to spend time working together at such centers.

There are two groups of questions that the experience with such centers should try to answer.

1. What happens when all the subsystems needed for this idea are plugged together? If fuses blow, what's the cause? Are there good fixes for the bugs?

[1] These include the Huntington simulation programs, the Denver CMCP booklets, Project REACT materials, and Project Solo curriculum modules. Materials are also available from computer vendors such as DEC, Wang, and HP. Two commercial publishers that have computer materials are Houghton Mifflin and Scott Foresman. Magazines with curriculum ideas include the People's Computer Company newspaper, Creative Computing, EDU (Digital Equipment Corporation), and the HP newsletter.

Once the mechanics of the system are ironed out, what about its value? Does case study evaluation hold up for larger groups?

2. Is the idea of a computer-based learning center which (like a museum) services several schools a viable one? What are the best time spans for attendance at such centers? If teachers from conventional schools run programs at the center part of the year, what is the effect on their regular teaching? On their students? On the curriculum? What do parents think of the arrangement? Administrators? Do these people recommend establishing a similar lab setting in their own schools? Is this decision different after the adults have had opportunities to use (not just visit) the labs?

These are all intriguing questions, suggesting as they do an approach to avant-garde innovation that builds (in fact depends) on existing school systems. The possibility that this approach would bring about real improvement in both the teaching and the curriculum of the participating schools, while also involving the adult community, bears further investigation.

Experience with These Ideas at Project Solo

Project Solo is an NSF supported program which has been exploring the use of computers in high school mathematics. It was started in 1969 with a small experiment involving about 50 students in one school. By 1972, several hundred students in each of three large public schools were involved in using project-produced curriculum modules to supplement standard courses in mathematics : nd several other subjects. The final project report (June 30, 1972) listed about a dozen pros and cons to this approach. On the positive side were such things as clear evidence that teachers and students could manage the technical demands of student controlled computing; teacher-parent-student enthusiasm (and pressure) to keep the project going after NSF funding terminated, and impressive new work by students which was accompanied by at-least-as-good performance on conventional tests.

Difficulties encountered included the problem of convincing school administrators that they should allocate lab funds (both capital and operational) to a subject (mathematics) traditionally associated with a textbook-only budget; handling the conflict between the rigidity of school schedules and the flexibility needed in pursuing creative project ideas; the problem of adequate guidance and direction, and the limitation that standard I/O devices impose on the way one thinks about mathematics.

In addition to computer extensions of standard curricula, one of the schools experimented with a new elective course called "Computer Science" in which an individual project approach was encouraged. One such project is documented in (14). The course ran into difficulty when it became apparent that student requests for enrollment in the second year would far outstrip facilities. The course was therefore put on hold. (A time-sharing computer centrally located in a new school scheduled for completion next year looks like a solution to this impasse.)

Project Solo is no longer directly involved with in-school work, but is exploring instead the third "other" option to curriculum development as part of what is now called the Soloworks Lab (15). This is a small operation at the University of Pittsburgh where we are developing the hardware, software and courseware to support a very open view of mathematics.

Our approach is to determine such things as content, sequencing and requisite skills in a top-down fashion; that is, on the basis of what we find a student needs in order to carry out a variety of creative project ideas. The computer is central to most of this work, being used to control what we call "event-worlds." The elements for these worlds include such computer peripherals as terminals, a robot, a pipe organ, graphic displays, a plotter, a flight simulator and a general assortment of controlled sights and sounds. The final report on this work is not scheduled until Fall of 1976. This report will include information about new hardware and courseware that may be of interest to schools with an experimental bent.

Summary

The word curriculum comes from the Latin *currere* meaning "to run." Whether students will still be running through today's course material 26 years from now (when the year 2001 becomes reality) is doubtful. That computer technology will influence the new curriculum is almost certain. The shape of that influence is going to be pretty much determined by what school people and researchers do with computers in the next decade. All the options discussed in this paper need to be explored carefully, and in real environments where the "common" sense of teachers and students can exert maximum influence. □

Bibliography

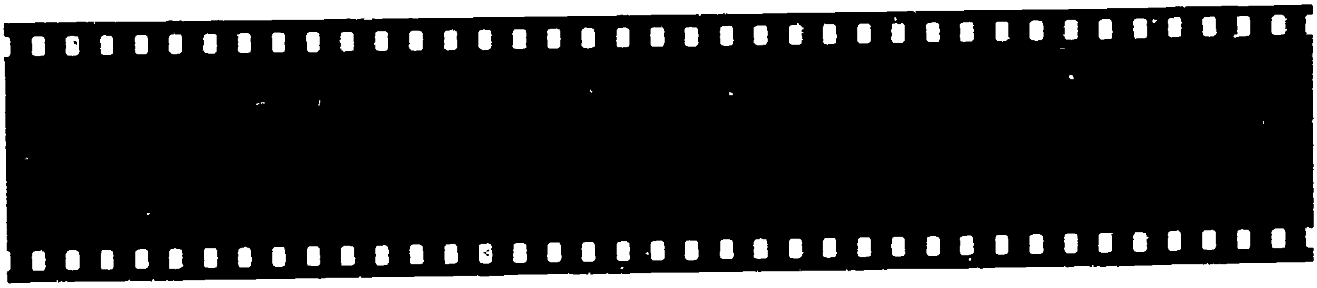
1. Dwyer, T. A., and Critchfield, M. D., "Computers and the Romantic View of Education", *Technological Horizons in Education*, Vol. 1, No. 3 (1974), pp. 4-9.
2. Berry, D. and Richards, S., "CLASSIC: Classroom Interactive Computer", *Technologicel Horizons in Education*, Vol. 2, No. 1, (1975).
3. Luehrmann, A., "Should the Computer Teach the Student, Or Vice Versa?", *AFIPS Conference Proceedings*, 40 (1972), pg. 40.
4. Dwyer, T. A., "Towards a Theory of Computer-Enhanced Learning", *Project Solo Working Paper #7508*, University of Pittsburgh, (1975).
5. Nievergelt, J., "Interactive Systems for Education—The New Look of CAI", *Paper at the IFIP 2nd World Conference On Computer Education*, September 1975, Marseilles, France.
6. Koetke, W., "Computer Facilities and Secondary School Mathematics", *Interface*, Vol. 5, No. 2, (April 1971).
7. Burleigh, M., "How Computer Programming Helps Students Become Mathematicians", *Creative Computing*, Sept.-Oct., 1975.
8. Bell, F., "Why is Computer-Related Learning So Successful?", *Educational Technology*, Vol. 14, No. 12 (1974), pp. 15-18.
9. Critchfield, M., "Trends in Mathematics Curiculum Reform, 1964-1974", *Creative Computing*, May-June, 1975.
10. Morton, A. K., "Project Compute", *Dartmouth Computer Center Brochure*, Hanover, NH, (1975).
11. Dwyer, T. A., "The Significance of Solo-Mode Computing for Curriculum Design", *EDU*, 13 (September 1974).
12. Dwyer, T. A., "Some Principles for the Human Use of Computers in Education", *Intl. Jnl. of Man-Machine Studies*, 3 (1971), pp. 219-239.
13. Dwyer, T. A., "Heuristic Strategies for Using Computers to Enrich Education", *Intl. Jnl. of Man-Machine Studies*, 6 (1974), pp. 137-154.
14. "My Computer Understands Me", 16 mm film distributed by DEC film library, Maynard, MA 01754.
15. Dwyer, T. A., "Soloworks: Computer-Based Laboratories for High School Mathematics", School Science and Mathematics, January, 1975.

SOLO WORKS

Project Solo,
University of
P`ttsburgh,
Pittsburgh, Pa.
15260 (412) 624-6461

Newsletter #33

December, 1975



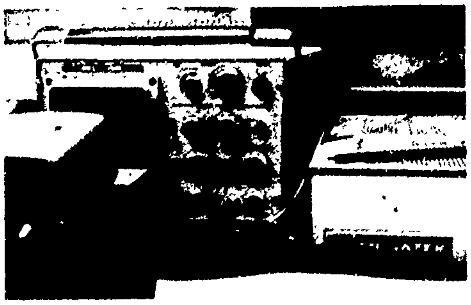
COMPUTER GRAPHICS AT SOLOWORKS

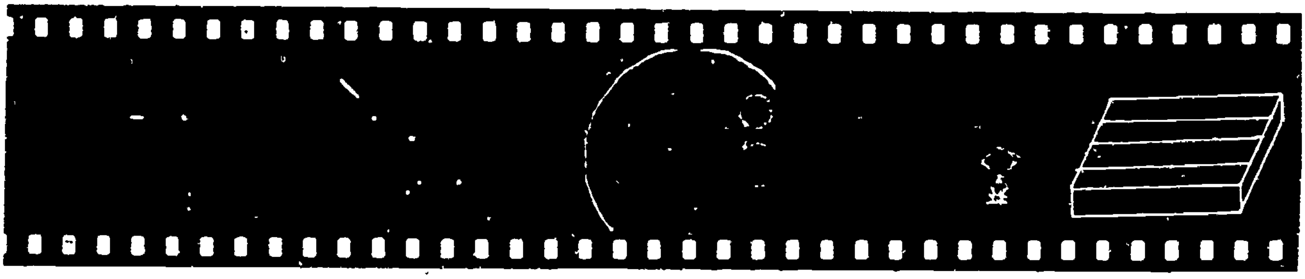
We've been working with quite a variety of technology at Soloworks, some of it off-the-shelf, some of it specially developed. The five "big" winners so far (in terms of both student interest and richness of content) are the RSTS 11/40 system, computer graphics, computer music, the Frasca flight simulator, and computer robots.

THE MEGATEK GRAPHICS SYSTEM

Among the various computer graphics systems now available, one of the most interesting is made by a company called Megatek (1055 Shafter St., San Diego, CA 92106). It's a winner on several counts: low cost, ease of programming, high reliability (100% for us), and a company that bends over backwards to help its customers. The graphs above were made on this system, as well as the other pictures shown at the top of the next page.

The Megatek unit is the small box shown at the right side of the photo below. The version we're using connects with just two wires to any serial port on your computer (usually in parallel with a terminal which remains usable). There are other versions that slip into a slot inside your computer. The output is displayed on an oscilloscope with X-Y inputs. We've had lot's of fun "broadcasting" this picture all over our lab by simply focusing a cheap TV camera on the oscilloscope screen and connecting the camera to low cost TV monitors. The picture at the top of this page shows one of these monitors sitting in our PDP-11 rack. For more examples of Megatek Graphics output, turn the page ⇨





Frames 1 and 2 above show two snapshots of a many-bodied planetary motion program in action, complete with a sun, planets, satellites, and comets. Frame 3 shows a space craft orbiting around a moon with non-homogeneous mass (ala swiss cheese). Frame 4 is the classical trajectory problem with a few extra surprises, while frame 5 shows a swimming pool in which realistic multi-event meets are simulated.

UPDATE FROM THE SOLOWORKS LAB

We've had quite a few inquiries asking whether any Soloworks newsletters have been mailed since #27. The answer is no, mostly because our small (albeit dedicated) staff has been swamped with many other tasks, ranging from the development of new curriculum ideas and new technology, to the running of a "mini-school" to find out what kind of work kids can do in such an environment. We've also had to economize by consolidating mailings of newsletters, since adding new staff to keep up with the growing demand isn't possible at this time.

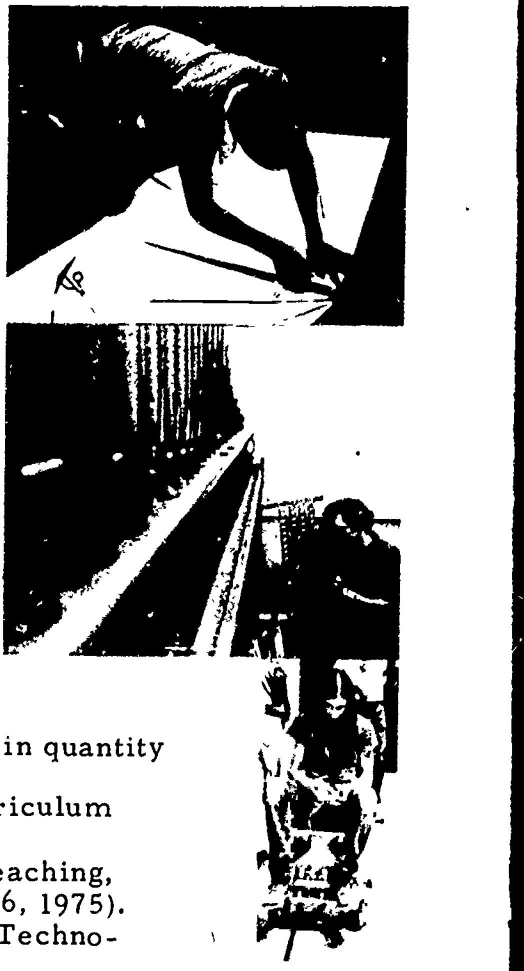
The three issues in this December mailing report on some of our work with the curriculum aspects of a computer-lab approach to mathematics. A sample curriculum module entitled "Art and Mathematical Structure" will (probably) appear in Creative Computing in 1976 (as a replacement for newsletter #34), while a beginning answer to the question "What next? " will appear in a later Creative Computing issue under the title "The Art of Education: Blueprint for a Renaissance" (Soloworks newsletter #35).

SUMMARY OF NEWSLETTER MAILINGS

1. Project Solo Newsletters ran from #1 to 22. These are now all out of print. Complementary materials can be found in the two books, A Guided Tour of Computer Programming in BASIC, and Computer Resource Book: Algebra. These can be ordered directly from Houghton-Mifflin, One Beacon St., Boston, MA 02107.

2. Soloworks Newsletter #23, 24, 25, 26, and 27 were mailed in 1974, and these are now out of print.

3. Soloworks Newsletters #28, 29 and 30 were not printed in quantity since they appeared in other publications as follows:
   #28--The Significance of Solo Mode Computing for Curriculum Design, EDU #13 (September, 1974).
   #29--Some Thoughts on Computers and Greatness in Teaching, SIGCUE Annual Computer Topics, (ACM, NY 10036, 1975).
   #30--Computers and the Romantic View of Education--Technological Horizons in Education Vol. 1, No. 3.
   #31, 32, 33--are enclosed as our December 1975 mailing.



Photos above: Designing a paraboloid, a computer controlled organ, and part of a robot.

THE CASE FOR A GENERATIVE CURRICULUM

The word curriculum has traditionally meant a pre-determined structure developed by "experts" for the universal benefit of large masses of students and their teachers. This turns out to be an idea of limited value simply because there is much more to learning than such an expert-to-teacher-to-student "transmittal" model takes into account. In particular, the use of "experiential" and "creative" approaches to learning can multiply the effectiveness of "transmittal" techniques many fold.

One way to describe our efforts at Soloworks is to say that we have been looking for good ways to sensitize students as good receivers. We have been working to develop environments where students will say (as one recently did), "I can't wait to take that course in the theory of how transistors work now that I've used them to do so many neat things". That student is clearly ready to get a lot out of a fixed curriculum, even a mediocre one.

But why accept mediocrity? The development of good "receivers" means that the curriculum transmittal process can become a whole new ball game. It's now feasible to "transmit" more complex signals, some carrying very advanced information, because:

(a) good receivers have the capability for selective tuning, that is, lots of local/personal adaptation and change, and

(b) because a sensitive (creative) receiver can extract information from what might be a "noisy" signal for others.

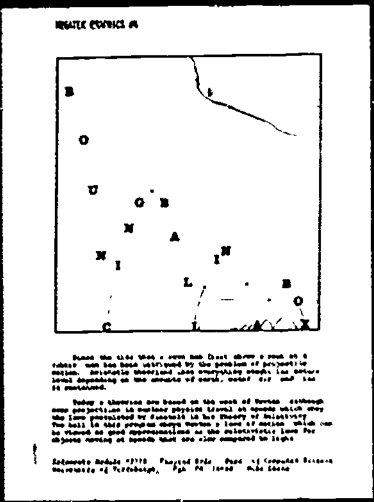
The curriculum modules we have developed at Soloworks have been based on this viewpoint, and so some of them may appear to be advanced. But most are really quite accessible, if one's receptive powers have been enhanced by the chance to first do things with the ideas, especially on a computer.

Another way to make advanced ideas accessible is to organize them in such a way that the beauty of the classical ideas from which they derive is more obvious. The scheme at the right represents one such organization. It can be thought of as a curriculum superstructure that ties computer lab work and theory together. However, it's meant to be a very flexible structure. We expect that teachers and students will modify it (sometimes considerably), and that it's main function will be to set their sights high.

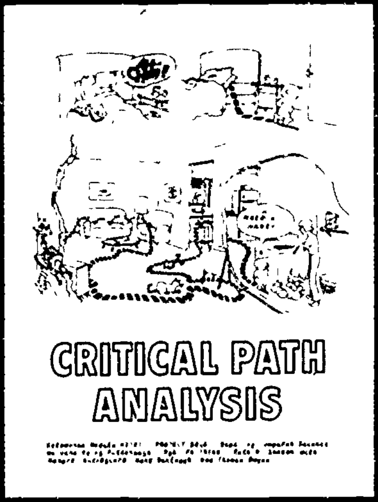
REPORT ON THE PDP 11/40 RSTS SYSTEM AND BASIC-PLUS

We have been using this system now for a year and a half without a single day of down time! And this is with a constant flow of users who bring the system up and down, modify it, mess with it, and ask it to do all kinds of wild things. BASIC-PLUS continues to inspire better and more sophisticated uses. All in all, it's an excellent product, with both the reliability and sophistication education deserves.

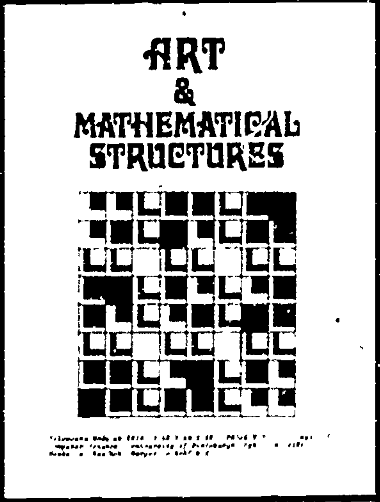
NEW MAGAZINE--We recommend you take a look at BYTE, a magazine dedicated to the "personal" computer movement. Subscriptions are $12.00 from Green Publishing Inc, Peterborough, New Hampshire, 03458.



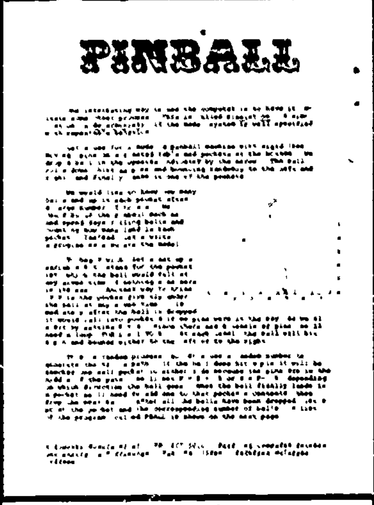
MEGATEK GRAPHICS #1



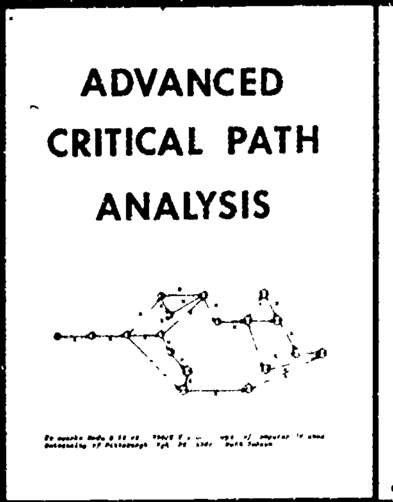
CRITICAL PATH ANALYSIS



ART & MATHEMATICAL STRUCTURES

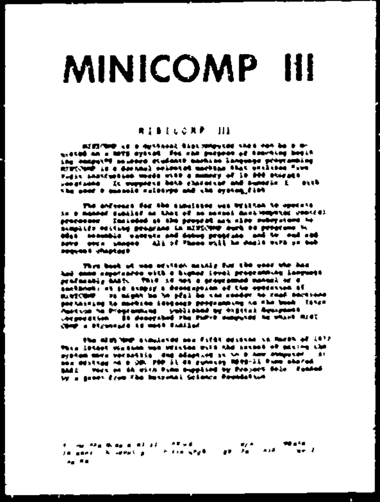


PINBALL



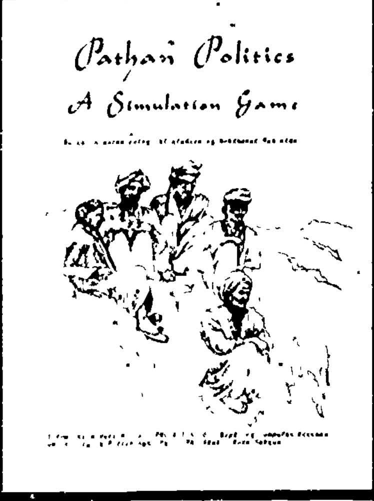
ADVANCED CRITICAL PATH ANALYSIS

Multi-server Multi-queue systems



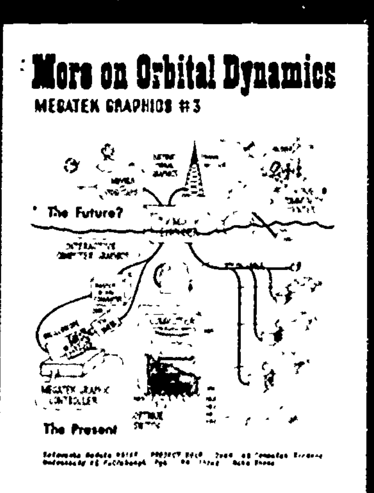
MINICOMP III

MINICOMP III



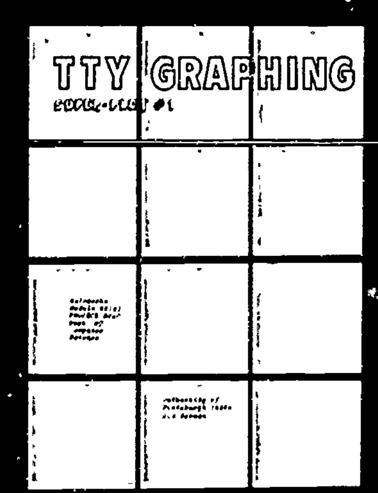
Pathan Politics

A Simulation Game

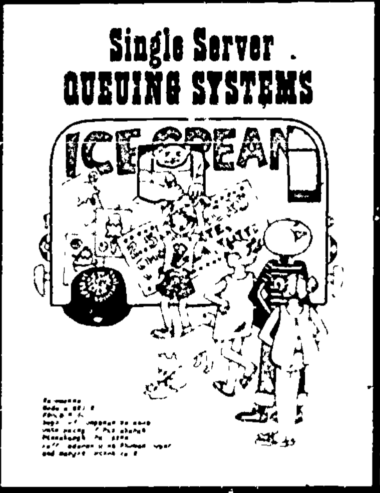


More on Orbital Dynamics

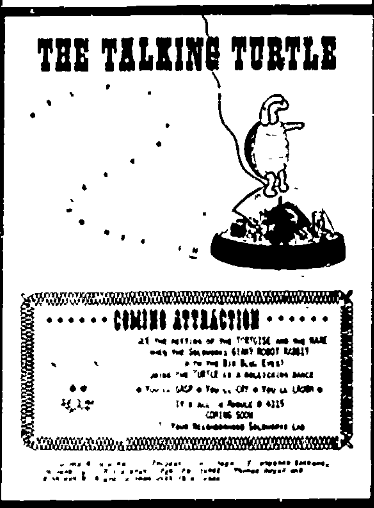
MEGATEK GRAPHICS #3



TTY GRAPHING



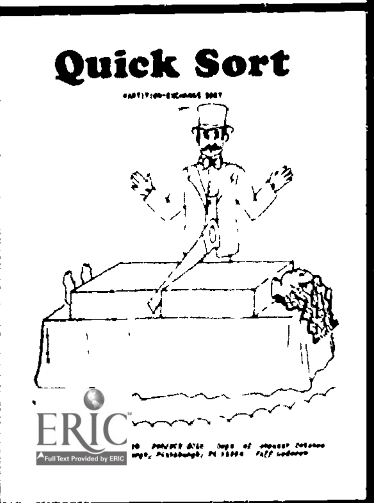
Single Server QUEUING SYSTEMS



THE TALKING TURTLE

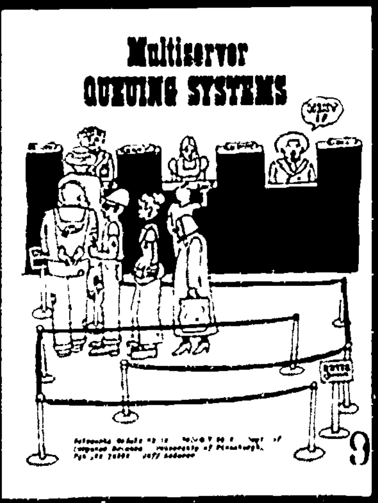
COMING ATTRACTION

COMING SOON

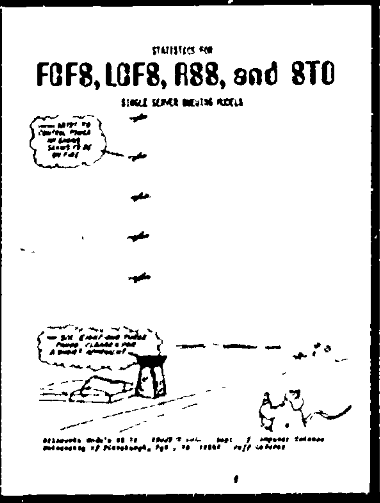


Quick Sort

PARTITION-EXCHANGE SORT



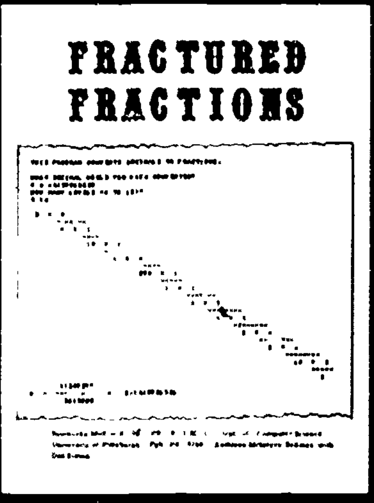
Multiserver QUEUING SYSTEMS



STATISTICS FOR

FCFS, LCFS, RSS, and STO

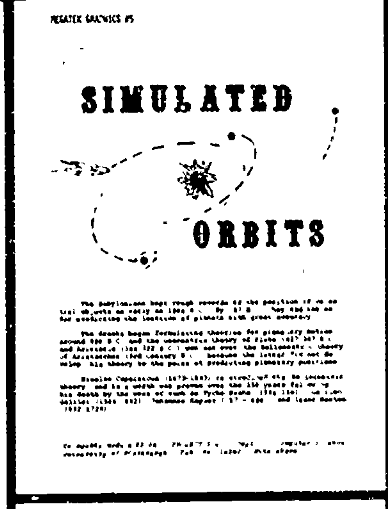
SINGLE SERVER QUEUING MODELS

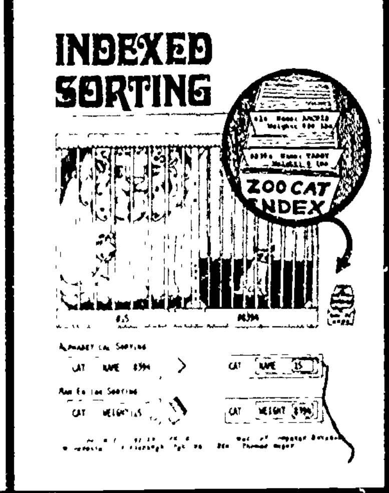


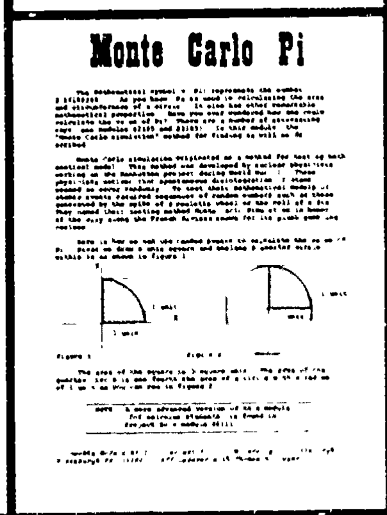
FRACTURED FRACTIONS

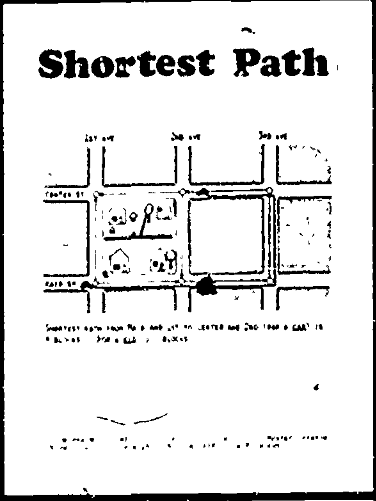
SAMPLE MODULE TOPICS

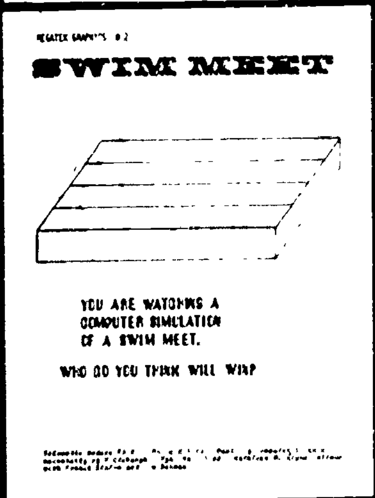
The curriculum structure described on pages 3 and 4 is based on the availability of modules which can be re-arranged in various ways. The pictures to the right are reproductions of the covers of some of the modules we have developed at Soloworks. Since most of these modules are rather long (and also because they are experimental) we are unable to print them in quantity for distribution. If any kind of arrangement can be made for distribution we'll let you know.

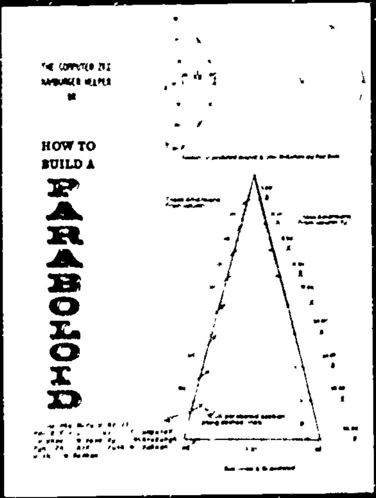


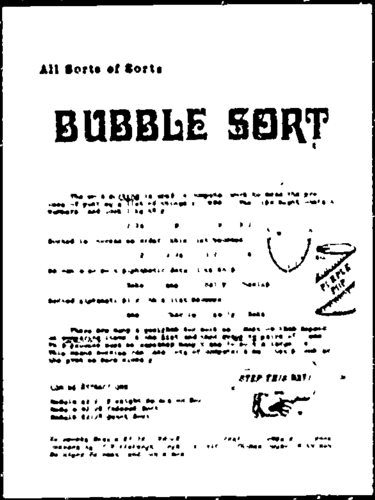


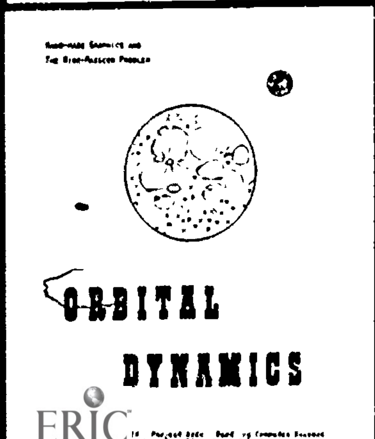


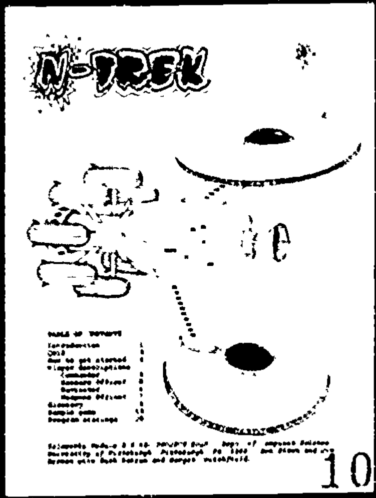


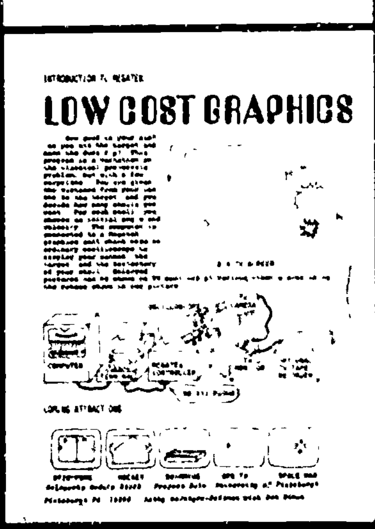


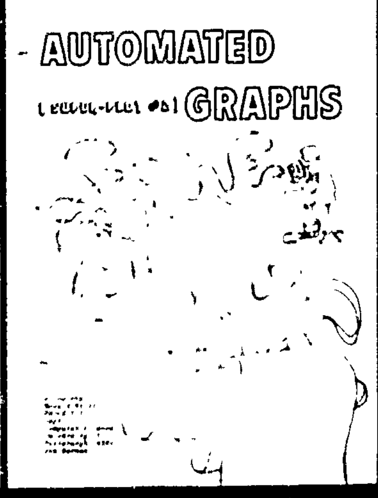












Example of a Computer-Related Math Curriculum

Sample Course Units	*Classical Ideas*	*Related Curriculum Modules*
THE PREP SERIES		
A1 A Whirlwind Tour of Computer Programming	Programming	A Whirlwind Tour of BASIC
	Variables	A Guided Tour of BASIC;BASIC-PLUS Tutorials
A2 Professional Secrets of Exact Arithmetic	Operators	Intense Addition:Intense Multiplication,Foreign Currency Conversion Prog.
	Arithmetic	*Grouping Tricks;Inverse Tricks;Metric Conversion
A3 Computer Arithmetic: Absolute and Relative Error	Error Analysis	Supermarket Estimation.A Home Accounting System
	Number Theory	How Wrong Is Wrong?:Multiple Precision Arithmetic
A4 Professional Secrets of Approximate Arithmetic	Distributive &	Fantastic Fractions,Rational Approximation
	Associative laws	*Impossible Problems Made Possible
A5 Beyond Arithmetic: A Soloworks Sampler	Spiral	*Computer Lab Preview;Dynamics Lab Preview
	Curriculum	*Synthesis Lab Preview;Modeling Lab Preview;Critical Path Analysis:I
THE COMPUTER LAB SERIES		
C1 Computers, Algorithms. and Game Theory	Algorithms	NIM;Vector Race: Horse Race:Crazy Eights,Tennis;Star-Trek;
	Coordinate	Submarine;Baseball;Basketball;Football:*Swimming:Bowling
C2 Computer Graphics	Geometry	Simple Teletype Graphics;Plotters:Polynomial Plots:
	Polynomials	Horner's Algorithm:*Budget CRT Graphics:*The Megatek System
C3 The Power of Algebra: Finite Algorithms	Linear & Quad Eq.	Linear Systems;Gauss 2;Gauss 3.Gauss N;Quadratic Solver.
	Linear Systems	Quadratic Coding;GCD;Primes:The Big Ear:Computer Design of Paraboloids
C4 Iteration; Infinite Algorithms	Nonlinear Eq.	Root-Finding;Binary Search.Convergence Tests:*Secant Method
	Iteration; Roots	Synthetic Division:Newton's Method:Dwyer's Method:Finding Derivatives
C5 Graphing Multi-Valued Functions; Arrays: Sorting	Matrices	Picture Arrays:Parametric Equations
	Order Relations	Sorting Tricks;Lissajous Figures:Polar Plots
EC6 Data Structures: Trees. Stacks, Queues	Post-Fix Arith.	Enumeration Problems:Reverse English.Polish Notation
	Simulation	*Mobiles:Circular Queues;33 Flavors
EC7 Recursive Programming	Recursion	Factorials:GCD Revisited Recursively
	Polar Coord.	*Turtle Geometry;Critical Path Analysis:II
EC8 Computer Organization	Computer	Simulating A Computer;Machine Language
	Literacy	Minicomp III;Computer Generated English,Russian,Greek Alphabets
EC9 Compilers and Interpreters	Languages	How to Write a Compiler
	Operating	How to Write an Interpretor
EC10 Systems Programming	Systems	Mailbag;System Accounting;A File Fornating System,
		Interactive Editors:Report Generators:Computer Generated Ditto Masters
THE DYNAMICS LAB SERIES		
D1 Geometry, Time, and Motion	Distance	*From Euclid to Newton to Einstein
	Metric Spaces	Bounce Animation;Basketball Trajectory;Evel Knievel
D2 Integrating Machines	Integral	Mechanical Integrators;Analog Integators
	Calculus	*Digital Integrators;3 A/D, D/A Lab Projects
D3 Conquest of the Sky: VFR Flight	Kinematics	Principles of Flight
	Vectors	How to Fly en Airplane
D4 On Solid Instruments: IFR Flight	Trigonometry	*How to Navigate an Airplane
	Anal. Geometry	*Instrument Landing Systems
D5 Moon Landing	Differential	1-D Lunar:*Difference Equations
	Calculus	2-D Lunar;*Fancy Lunar
ED6 Flights of Fancy: N-Trek	Diff. Tables	Star-Trex:Space Ship Gamma
	File Structures	N-Trek
ED7 Flights into Space: Orbital Motion	Kepler's	.Satellite Orbits
	Laws	Space Wer
ED8 Flights of the Mind: Mathematical Spaces	Euclid	*Euclidean Space;Vector Spaces
	Descartes	*Finite Abstract Spaces
ED9 Flights of Invention: Creating New Worlds	Lobachevsky	*Infinite Abstract Spaces
	Riemann	*Function Spaces: Functional Analysis
ED10 The Theory of Relativity	Einstein	
	Lebosgue,Hilbert	
THE SYNTHESIS LAB SERIES		
	Step Functions	Computer Representation of Synchronized Events
S1 The Mathematics of Orchestration	Fractions	A Music Compiler,Editor, and File System
	Random Numbers	Computer Composition of Music;Rounds and Harmony
S2 Music in the Air: Pipes and Strings	System Design	How to Build an Orchestra-to Computer Interface
	Binary Codes	*How to Build Computer Controlled Instruments
S3 Electronic Music: Synthesizers and Filters	Harmonic Serie	*The Natural Richness of Natural Music
	Fourier Series	Synthetic Music and the Formant Theory
S4 Stereo Systems: Design and Measurement	Wave Geometry	*The Secrets of Professional Stereo
	Logarithms	*Fine Tuning Your Stereo: The Power of Measurement
S5 Quedraphonic Sound: Codir[illegible] Decoding	Matrices	*Four into Two: Matrix Coding
	Information	*Two into Four: Matrix Decoding
ES6 Multi-Media Worlds: The G[illegible]y of Projection	Theory	Computer Control of Multi-Modie Technology
	Geometric Optics	*Lenses, Mirrors, and Pictures Made of Light; *Laser Geometry
ES7 Abstract Orchestrations: Mathematical Approximation	Fractions	Continued Fractions;Polynomial Interpolation
	Weierstrass	Mini-Max Approximations;Famous Infinite Series
ES8 Functions and Transformations	Tchebychef	*Relations and Functions
	Topology	*Mappings and Transformations
ES9 The De-Orchestration Problem: Statistical Analysis	Probability	*Predicting Football Scores, Elections,And
	Weighted Data	Other Logic-Defying Events
ES10 Cryptography	Permutations	Enciphering with the One Time Pad
	Statistics	*The Deciphering Problem
THE MODELING LAB SERIES		
M1 Single Equation Models	Interpolation	Linear and Nonlinear Predictors
	Functions	Policy Models;Conflict Models
M2 Logical Models: Truth Tables	Boolean Alg.	Digital Logic: 3 Lab Exercises in Digital Interfacing
	Number Bases	*Macro-Micro Logic
M3 Systems of Equations: Relaxation Methods	Partial Diff.Eqs.	Kirchoff Models:The Dirichlet Problem
	Green's Functio-	Random Walk Models
M4 Grephs, Networks, and Boundary Value Models	Matrix Polyn.	Communication Networks
	Relaxation Mthds.	PDE&Nonlinear-Networks; Relaxation Methods
M5 Analog and Hybrid Models	Digital Design	*Multi-Screen Multi-Terminal Systems; *How to use TV with Computers
	Abstract Spaces	*Low-Cost Color Graphics for Minicomputers
M6 Dynamic Models: Systems of Differentiel Equations	Perturbations	*Advanced Orbital Mathematics
	Coupled D.E.	*Inside Flight Simulators
EM7 Geometric Models: Crystallography	Plane Geometry	*Snowflakes
	Solid Geometry	*Crystelline Architecture
EM8 Finite State Models	Automate	Finite State Automata
	Psychology	Models of Behavior
EM9 Statistical Models	Statistics	Monte Carlo Models
	Sociology	*Data Based Models
EM10 Adaptive Models: Optimization	Linear Prog.	*Mathematical Programming;Dynamic Programming.
	Gradient Mthds.	*Optimal Seeking Methods
EM11 Feedback Models: Cybernetics	Complex Nos.	*Feedback Control
	Maze Algorithms	*Robots
EM12 Models of Intelligence	Natural	Artificial Intelligence;Semantic Nets
	Language	Computer Programs that Learn;Hex
[illegible]i-Mode Models	Mathematics	Anthropological Models
	Applied to	Simulation Games; *Ecological Models
[illegible]rective Models: Computer Animation	Non-determin-	An Interactive Nonlinear-Network Model
	istic systems	*Advanced Computer Graphics